

LAVA + OpenQA = Automated, Continuous System Testing

Laurence Urhegyi & James Thomas



Introductions

- Laurence Urhegyi
 - Project Manager
 - laurence.urhegyi@codethink.co.uk
- James Thomas
 - Senior Engineer
 - james.thomas@codethink.co.uk

Both have a wealth of experience in the Automotive Industry, and are passionate about FOSS



Contents

- Long Term Maintainability
 - What is this and why do we care?
- What is the problem?
 - Upstream / downstream gap for LTM
- How we can help towards solving it?
 - Start testing in upstream directly
 - Heavy automation of testing
- What have we used?
 - LAVA
 - OpenQA
- What we have done?
 - Case study 1: openQA in GNOME upstream
 - Case study 2: LAVA/openQA in kernel testing
- Future plans

The problem



Why Long Term Maintainability?

Many software systems / devices / products are intended to function over a long-term lifetime.

They rely on open-source components.

Software complexity is increasing.

More features = more loc = more bugs = more testing is needed.

The cost and complexity of maintaining such systems will continue to grow over time.

Organisations need to be mature enough to handle increasing complexity and volume, whilst remaining competitive.

Don't you mean Long Term Maintenance?

- Well, yes and no...
- Typical scenario:
 - Organisations rely on components (such as the kernel) that are supported for less time than the lifecycle of their product
 - Backporting from upstream is required for updates
 - Full system upgrades are rare or avoided
- The status quo is to 'keep systems on life support'

We can do better...

Long Term Maintainability (again)

- Systems designed from the outset to:
 - Be as close as upstream as possible
 - Integrate new upstream with minimal overhead, reduce local patch carrying
 - Have robust deployment mechanisms in place, to roll-back if something goes wrong (atomic upgrades)
 - Have processes that allow you to know how to repeat a build done in the past, and also keep track of every component version deployed
 - **Have robust testing pipelines that makes you comfortable to update**

Testing, testing, one two, one two

- Everyone wants the same thing: stable software, including the latest features, for the long term
- To achieve this you need to have confidence in your software and confidence in the software you depend on
- This has to come from testing



Credit: [SoundCloud](#)

What do upstream propose?

- Always use the latest!
- <http://kroah.com/log/blog/2018/01/06/meltdown-status/>
- Makes complete sense... BUT

What downstream says

- Kernel is support for only 2 / 6 years
 - We need to support this product for 10 years
 - We do not want to support own own kernel fork
 - We are not confident enough upgrading to next kernel version is not going to cause regressions!
-
- Not every organisation is willing / able to commit resources for testing that allows for such long term maintenance
 - <https://lore.kernel.org/linux-arm-kernel/YCzknUTDytY8gRA8@kroah.com/>

Mind the gap!

- Testing modern software platforms is extremely complex and time consuming
- Companies often miss out on the latest fixes and features in Linux, for fear of the time-consuming manual regression tests required



What we can do to help?



Automated, continuous embedded system testing FTW!

- Increased automation improves robustness and transparency of tests
- Reduce manual efforts = increased testing capacity
 - Machines can be utilised 24 hours per day
- Catch major errors more easily, and earlier
 - Eventually increase focus on more complex testing
- This is not new:
<https://security.googleblog.com/2021/08/linux-kernel-security-done-right.html>

Intro to the tooling: LAVA & openQA



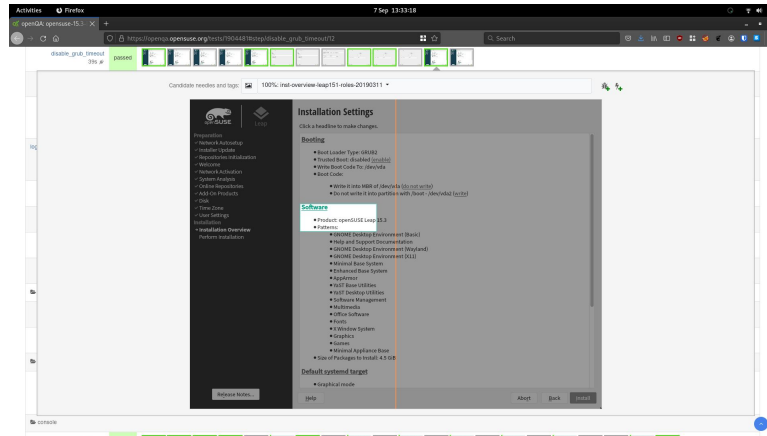
LAVA

- LAVA “continuous integration system for deploying operating systems onto physical and virtual hardware for running tests”
- Hardware orchestration for physical devices
- Central server that dispatches jobs to clients
- Used by KernelCI (<https://linux.kernelci.org/job/>) to test new kernel versions

[illegible]

OpenQA

- OpenQA for testing *software* (UI)
- Tests based on screenshot comparisons (Needles)
- Central server to view test results, and to dispatch test jobs to workers
- Workers provide a specific machine type (e.g qemu-x86_64)
- Tests performed how the majority of users actually use the software



Help upstream 1: openQA and GNOME



CI Integration

- Integrated into Gitlab CI pipeline
- Ad-hoc workers using the same build infrastructure, no “pets”, no dedicated test hardware needed
- Workers only need KVM

CI Integration

- Tests in the same git repo
- Different workers can run different tests if needed
- Needles in a separate repository
- Workers clone needles (OPENQA_NEEDLES_GIT, OPENQA_NEEDLES_SHA)
- Workers then register with OpenQA with a unique machine type (e.g. qemu-x86_64-12345) and start the test run previously built GnomeOS image

CI Integration

Activities Firefox 7 Sep 13:42:55

openQA Pipeline - GNOME / gnome-build-meta X +

https://gitlab.gnome.org/GNOME/gnome-build-meta/-/pipelines/314858

GNOME Menu Search GitLab Sign in / Register

gnome-build-meta

- Project Information
- Repository
- Issues 91
- Merge requests 22
- CI/CD
- Pipelines**
- Jobs
- Schedules
- Deployments
- Monitor
- Packages & Registries
- Analytics
- Wiki
- Snippets

WIP: gnome-shell patch

9 jobs for [jjardon/41rc](#) in 59 minutes and 2 seconds (queued for 7 seconds)

latest

1a2a4eee

1 related merge request: [I1295 WIP: GNOME 41rc](#)

Pipeline Needs Jobs 9 Tests 0

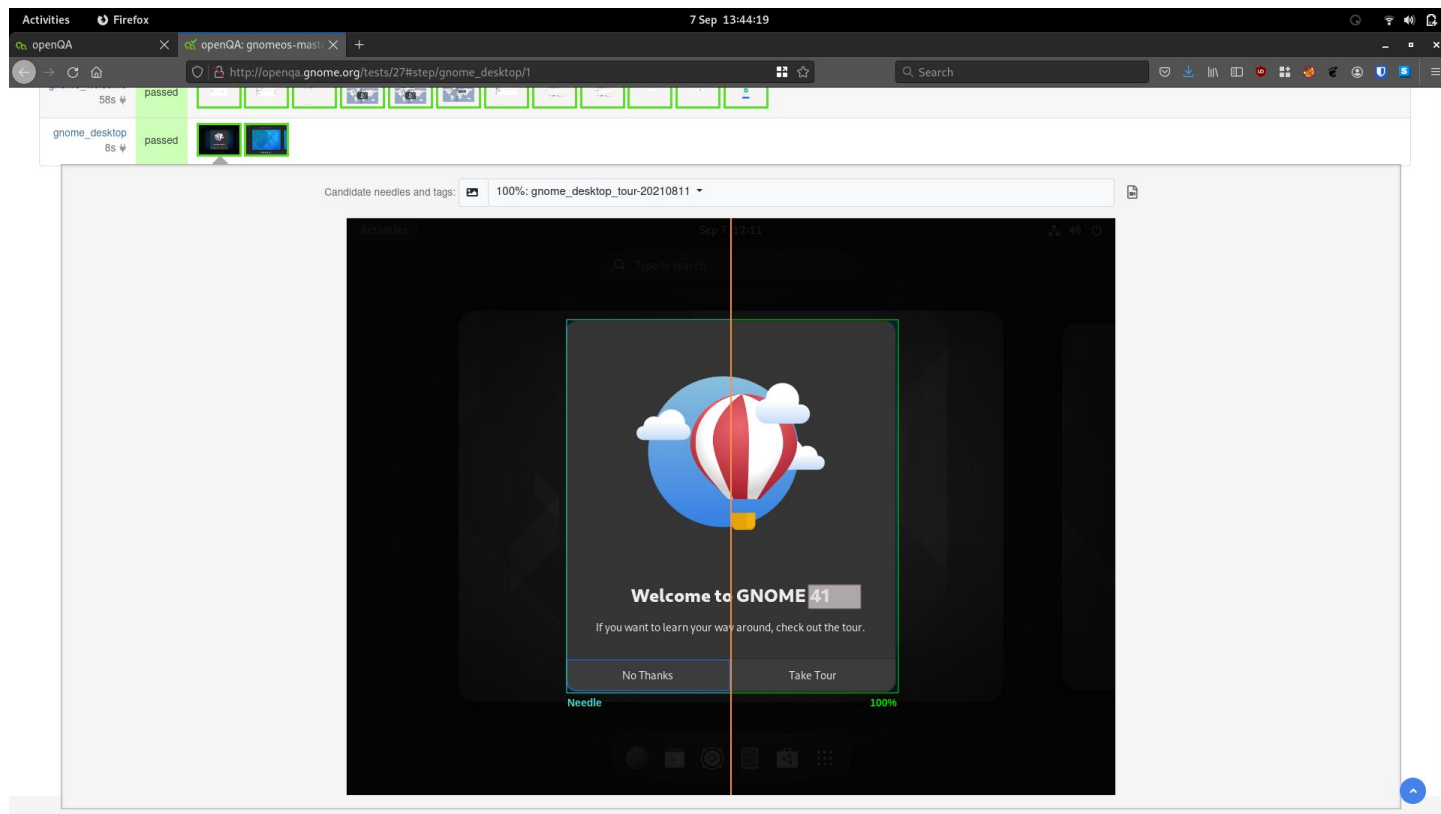
Group jobs by Stage Job dependencies Show dependencies

Tip: Hover over a job to see the jobs it depends on to run.

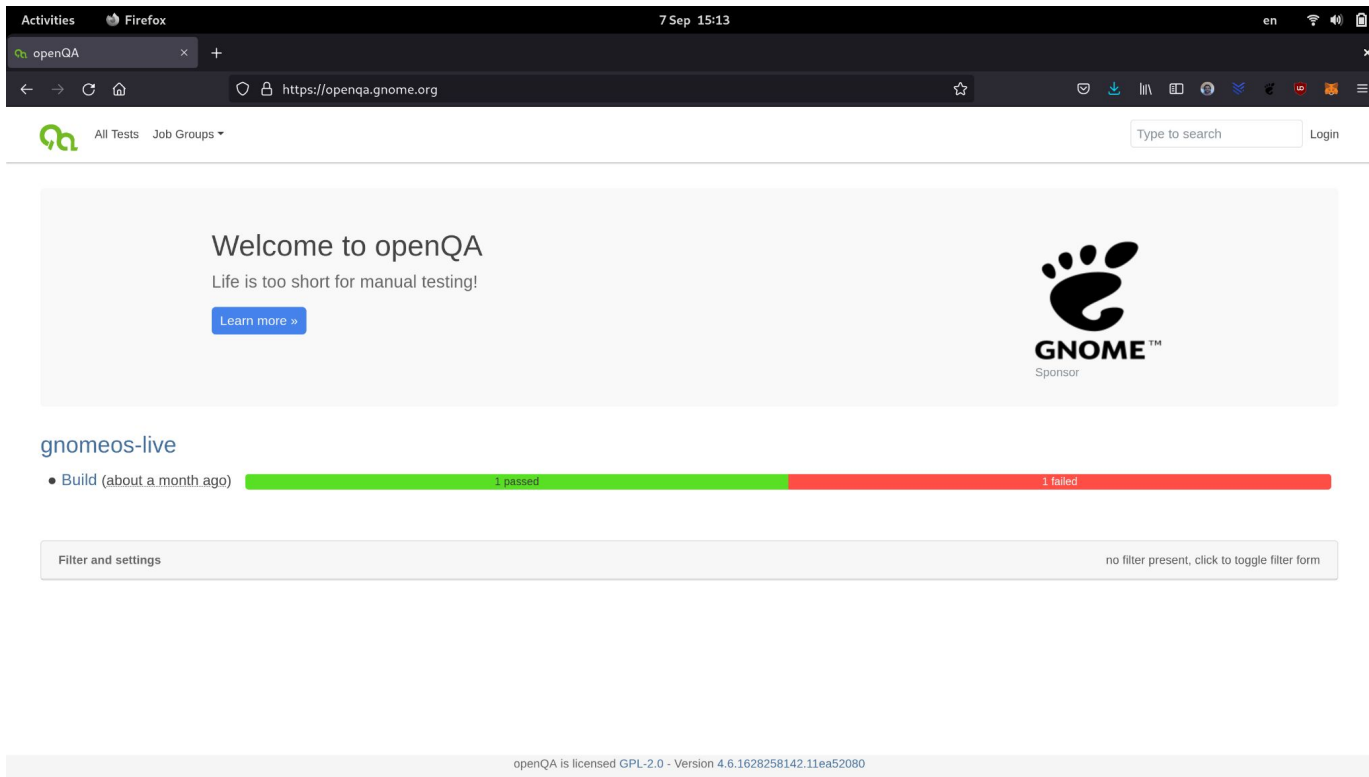
```
graph LR; build_aarch64[build-aarch64 build] --> deploy_group; build_i386[build-i386 build] --> deploy_group; build_x86_64[build-x86_64 build] --> deploy_group; subgraph deploy_group [ ]; pinebook_pr[pinebook-pr... deploy]; raspberrypi[raspberrypi... deploy]; rock64_image[rock64-image deploy]; vm_image_x[vm-image-x... deploy]; end; deploy_group --> test_iso_install[test-iso-install... test];
```

<< Collapse sidebar

CI Integration



Being used already: openqa.gnome.org



The screenshot shows a Firefox browser window with the address bar displaying `https://openqa.gnome.org`. The page features a dark theme and a navigation bar with links for "All Tests" and "Job Groups". A search bar and a "Login" button are also present. The main content area includes a welcome message, a GNOME logo with the text "GNOME™ Sponsor", and a section for "gnomeos-live" tests. A progress bar indicates that 1 build passed and 1 failed. A filter settings box is located at the bottom of the main content area.

Activities Firefox 7 Sep 15:13 en

openQA x +


← → ↻ 🏠 🔒 https://openqa.gnome.org ☆

🔍 All Tests Job Groups ▼ Type to search Login

Welcome to openQA

Life is too short for manual testing!

[Learn more »](#)

 **GNOME™**
Sponsor

gnomeos-live

- Build (about a month ago) 1 passed 1 failed

Filter and settings no filter present, click to toggle filter form

openQA is licensed GPL-2.0 - Version 4.6.1628258142.11ea52080

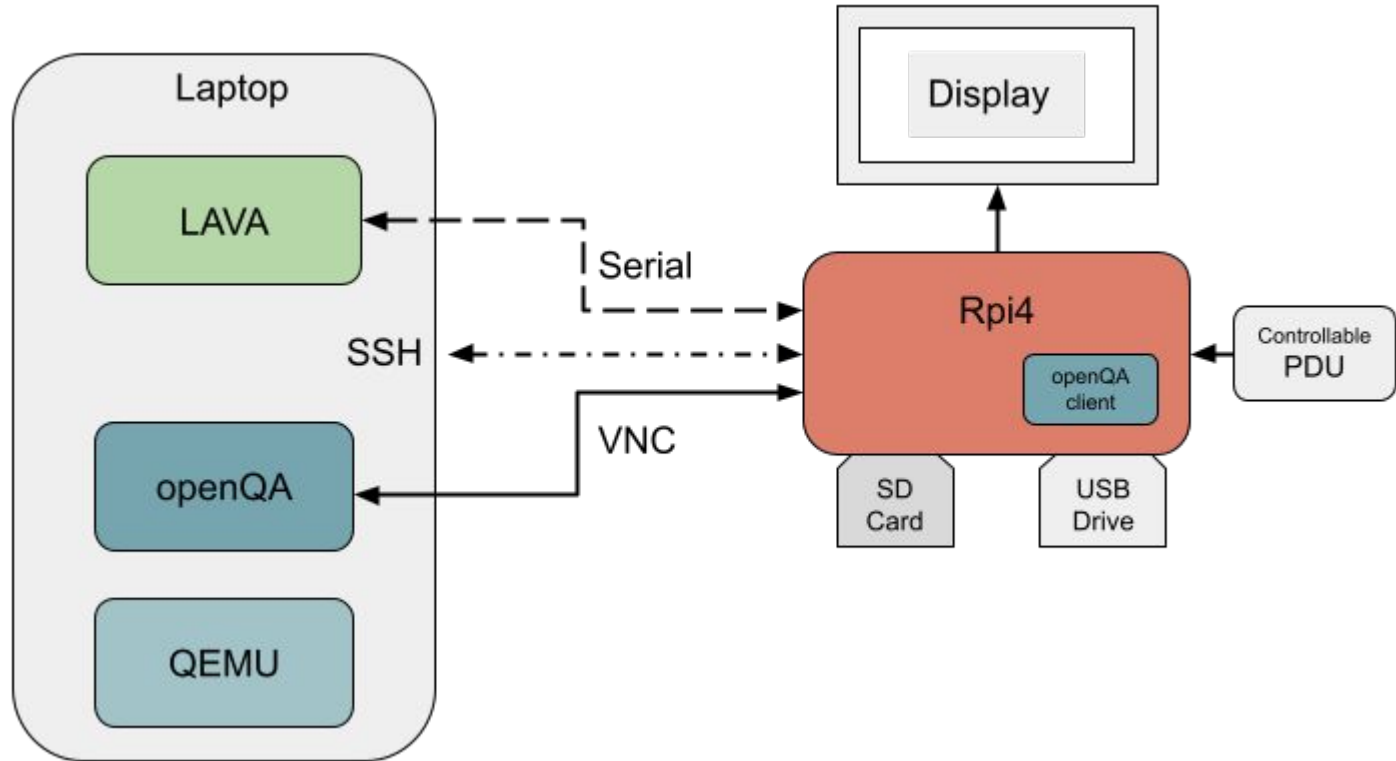
Help upstream 2: LAVA & openQA & Linux kernel



LAVA and OpenQA

- Testing on hardware rather than emulators
- Utilise the existing KernelCI tests
- OpenQA testing to check for regressions (e.g in the graphics driver)
- Check out the [blog post](#) for details of the setup

Architecture



It's being used!

- <https://lava.qa.codethink.co.uk>
- <https://openqa.qa.codethink.co.uk/>
- Positive feedback received
<https://lore.kernel.org/lkml/CAHk-=wjz+RhR8rr4rAZBPf-mxZXvn2RQe-XTQcL8X+HXBAFxBA@mail.gmail.com/>

What next?



Future Work

- More tests!
- Connect to KernelCI APIs
- Better Test/Needle synchronisation between workers and OpenQA
- Board deployment and testing added to Gitlab pipelines
- More backends! (e.g RDP)

Thanks for listening!

www.codethink.co.uk



Resources

- [GNOME testing blog post](#)
 - <https://openqa.gnome.org/>
 - https://gitlab.gnome.org/GNOME/gnome-build-meta/-/merge_requests/1251
-
- [Kernel Testing blog post: LAVA + OpenQA](#)
 - <https://lava.qa.codethink.co.uk>
 - <https://openqa.qa.codethink.co.uk/>